WHITE PAPER

attachmateWRQ™

Internet File Transfers: Security Holes and How to Fix Them

AttachmateWRQ • January, 2006

Contents

Introduction

As the demand for information sharing grows, the Internet has become the medium of choice for organizations that want quick, easy, and affordable file exchange. Most organizations today use File Transfer Protocol (FTP) to move private information across the public Internet. But FTP poses serious security risks that cannot be ignored.

This white paper highlights the problem with FTP and outlines the ten essential features to look for in an FTP replacement. It describes five scenarios in which products based on the SSH and SSL/TLS protocols can ensure reliable and affordable file-transfer security. Finally, this paper tells how AttachmateWRQ file-transfer solutions ensure interoperability, confidentiality, and integrity for data in motion.

The Problem with FTP

In terms of exchanging information, the speed, ease of use, and affordability of the Internet is undeniable. For organizations, it is like having a pre-designed telecommunications infrastructure as their disposal. And FTP has become the preferred mode of transportation for moving files.

Because it is a standards-based protocol, FTP is highly interoperable. It is also fast, familiar, and easy to use. For these reasons, it has gained popularity among all types of organizations—from large and small businesses to government agencies and nonprofits.

But FTP has one inherent problem: it is dangerously nonsecure. Any information transmitted via FTP travels over the network in clear text, which means anyone with a sniffer can read it. Furthermore, logons to FTP servers require a user name and password, which also travel in unprotected clear text and can be easily grabbed by malicious individuals with just a hint of technical savvy. This lack of security is particularly risky for organizations governed by stringent regulations aimed at protecting sensitive data.

Ten Essential Features to Look for in an FTP Replacement

The ideal replacement for FTP would provide the protocol's strengths—speed, ease of use, and interoperability—as well as the requisite security required for public Internet travel. Equipped with these capabilities, you could easily replace traditional EDI, an expensive solution that requires leased lines and value-added network charges. You could also augment secure e-mail, another popular file-transfer mechanism that is limited by the size of the attachment it can deliver.

The ten essential features to look for in an FTP replacement are highlighted below:

1. **Industry-standard client connectivity**
   Because you cannot control what software your partners and customers use for their client connections, you want to choose a security solution that is readily available and based on industry-standard protocols. These features will simplify and cut the costs of data exchange. You will also want to allow for different styles of doing file transfer. In some cases, you will want a browser connection using an on-demand thin client. In others, you will need a command-line client that can be scripted for automation. Or you may opt for a Windows client with a graphical-user interface that makes file transfers easy for users.

2. **Data confidentiality**
   The Internet is a public network, so it is imperative that you scramble sensitive data to make it unreadable during transmission. To that end, you need a standard means of encryption, such as Secure Sockets Layer (SSL), Transport Layer Security (TLS, the newer version of SSL), or Secure Shell (SSH). These protocols all include a range of standard encryption ciphers that provide different strengths of encryption.

3. **Data integrity**
   People with malicious intent thrive on tampering with data—for example, changing account balances or purchase-order amounts. To ensure that bits remain intact as they travel from client to server or between servers, you will need some type of message authentication code. This code will also serve as a way to ensure message authenticity. In addition, you will want an automatic way to restart an interrupted transfer at the point where the connection was dropped.

4. **Authentication**
   To launch a secure session, users must be able to verify their identities and establish their rights to connect to a given server. They must also be assured that they are connecting to the right server. This two-way authentication should take place in a secure way—either via strong authentication or via user name and password encryption. Without a solid authentication method, intruders can easily gain access to private files and systems via sniffing, social engineering, or password guessing.

5. **Automation**
   Organizations often automate routine file transfers to save time and money. For example, large commercial businesses use batch processing extensively for back-office tasks; multiple files can be transferred in a batch when the load on back-office systems is light. These jobs are scheduled and typically run via scripts. The key is to make sure that the scripts are using secure mechanisms to upload data.

6. **Access control**
   Most organizations want to give users access only to specific directories on the internal network, which requires defining users and their access rights. Organizations also want to limit the number of open ports in their firewalls to minimize the chances of malicious traffic getting through.

7. **Auditing**
   If a client or server is compromised and data are retrieved illicitly, you will want to know what happened. By logging incoming and outgoing transmissions, auditing tools can help identify suspicious usage patterns, inadvertent or overlooked access, and mischievous or purposeful attacks. These tools are crucial to discovering both apparent and subtle forms of intrusion, making them a critical part of your security solution.

8. **Centralized management**
   In order to meet growing demands for information from whole new audiences of users, centralized management of the client side of the file-transfer process is essential. In fact, it is the only effective way to tackle today's fast-changing configuration and deployment requirements.

9. **Cross-platform support**
   Your customers or partners may be running on platforms that are different from your own, which is probably heterogeneous. For that reason, the ideal secure file-transfer solution will provide cross-platform support.

10. **End-to-end security**
    Moving a file securely across the Internet is one thing; getting that file securely to the end point (the server that it updates) is another. The vast majority of financial fraud is the result of internal hacking rather than external penetration of a secure network, so the last leg of protection, from the DMZ to the back office, is critical. To guard against these inside jobs, you need end-to-end security.

Careful consideration of these issues will help you choose the safest possible solution for data exchange—and save you time and money in the long run.

Five Secure Data-Exchange Scenarios

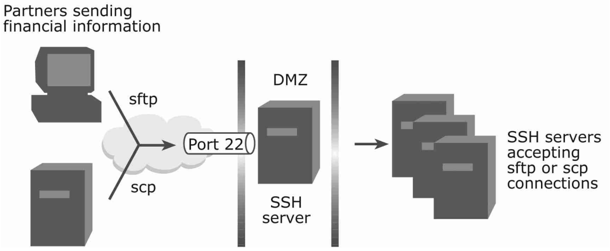
Organizations have different secure data-exchange needs. The following five scenarios show how standards-based secure file transfer can provide a low-cost and reliable way to protect data in motion.

Scenario 1: Financial services company

This scenario involves a financial services company that needs to transfer account information, conduct debit and credit transactions, and send check images.

The company's security solution is to run SSH File Transfer Protocol (sftp) and secure copy (scp)—two secure file-transfer mechanisms that are part of the SSH protocol—on a gateway or bastion host in the DMZ. This configuration allows limited application-level access between the external and the internal networks.

The sftp transfers are interactive, but scp transfers are automated so no user interaction is required. Authentication is host-based or accomplished using public keys with agent forwarding. (Another way to use the gateway host is to use the proxy-command feature to tunnel the sftp session from the gateway to the back-office servers.) Just one port—SSH port 22—needs to be open in the firewall to let these transfers pass through.

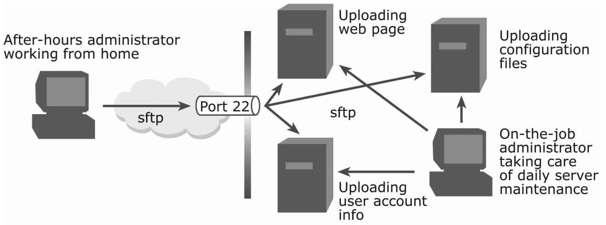

Partners sending financial information
sftp
scp
Port 22
DMZ
SSH server
SSH servers accepting sftp or scp connections

Scenario 1: Partners sending financial information

Scenario 2: IT organization

In this scenario, system administrators need to transfer configuration files, change user accounts, monitor usage records, and create new web pages.

Because administrators have root access to servers and often work over the Internet, they must protect their user names and passwords. By replacing FTP with sftp, the administrators can ensure that their user names and passwords are encrypted as they cross the wire. Alternatively, the administrators can use a stronger method of authentication—user public keys—to replace their passwords.

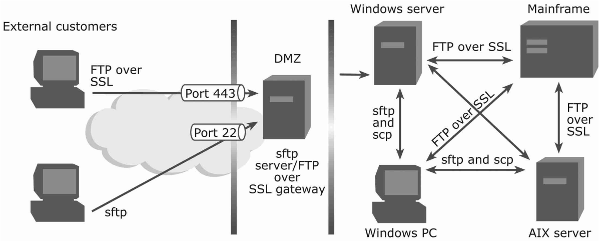

After-hours administrator working from home
sftp
Port 22
Uploading web page
sftp
Uploading configuration files
On-the-job administrator taking care of daily server maintenance
Uploading user account info

Scenario 2: After-hours administrator working from home

Scenario 3: Health care organization

This scenario centers on a health care organization that moves patient records internally, from server to server, and exchanges them with external entities.

After an external audit found passwords and sensitive patient information traveling across the network in plain text, the organization received a mandate to shut down FTP. Instead of FTP, which was used both internally and with partners, the organization adopted the SSH and SSL protocols. Now, partners can choose to use either protocol to the DMZ. Internally, the protocol of choice depends on which one is best suited to the relevant server environment.

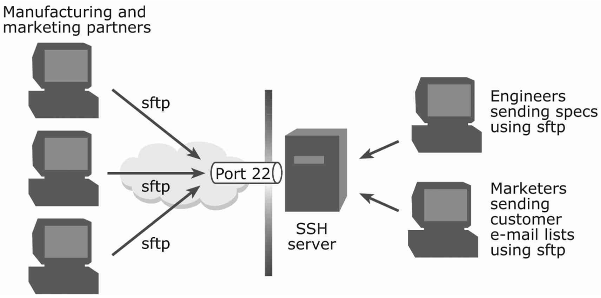


Scenario 3: A health-care organization moving patient records internally and externally

Scenario 4: Manufacturing company

In this scenario, a manufacturing company in a highly competitive industry has a number of partners, both for building its products and for marketing them. The latest design specs and customer lists represent intellectual property that needs to be protected, particularly because the industry is known for industrial espionage.

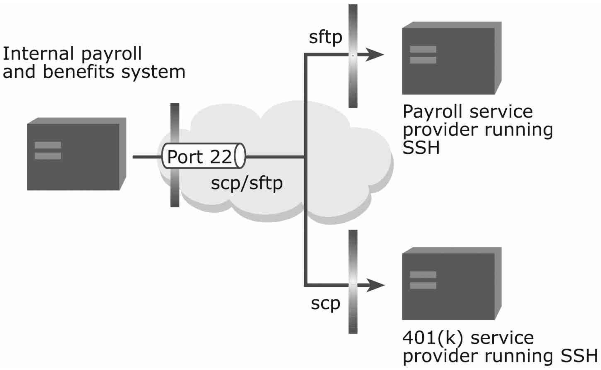
By using the SSH protocol suite and setting up one server behind the firewall for all sensitive file transfers, the company can monitor and safeguard all confidential communications.


Manufacturing and marketing partners
sftp
sftp
sftp
Port 22
SSH server
Engineers sending specs using sftp
Marketers sending customer e-mail lists using sftp

Scenario 4: A manufacturing company exchanging intellectual property with partners

Scenario 5: Organizations using services providers for payroll and health insurance

Whenever organizations exchange employee data (such as personal identification numbers, insurance claims, and medical information) with third-party providers, electronic files must be kept secure. SSH is a good fit here. With scp, file transfers can be easily automated.


Internal payroll and benefits system
Port 22
scp/sftp
sftp
Payroll service provider running SSH
scp
401(k) service provider running SSH

Scenario 5: Organizations using service providers for payroll and health insurance

AttachmateWRQ: Securing Data in Motion

For secure file transfer, AttachmateWRQ offers a variety of client and server solutions that are all based on the industry-standard SSH and SSL/TLS protocols. This section describes the secure file-transfer capabilities of these protocols and how they fit into the AttachmateWRQ security solution set.

File transfer capabilities in SSH

The main purpose of SSH is to transmit data over network connections using strong encryption and authentication methods. SSH is a replacement for nonsecure Telnet, FTP, X11, and Berkeley r-commands (rlogin, rcp and rsh)—all of which transmit data in the clear.

Currently, there are two versions of SSH: SSH1 and SSH2. The two versions are based on two distinctly different protocols, and they're not compatible. Also, SSH1 has been deprecated and is not recommended for use. For our purposes—securing file transfers—we are dealing with SSH2, which has been standardized through the Internet Engineering Task Force (IETF).

In SSH, the rcp (remote copy) command-line utility is replaced by the scp command-line utility (rcp has traditionally been used in UNIX environments for copying named files and directories). FTP is replaced by sftp, which provides all the functionality of FTP without the risks. Written as a subsystem for SSH2, sftp encrypts every bit—user names and passwords, directory listings, and files—exchanged between an sftp client and server. With the sftp capabilities in SSH, organizations can turn FTP off and eliminate that vulnerable point in the network.

SSH uses strong encryption ciphers, such as 3DES and AES, for data privacy, it also uses hashed message authentication code (HMAC) algorithms for integrity checking.

File transfer capabilities in SSL/TLS

SSL was developed to secure transactions over the web. It provides optional certificate-based authentication and secure data transfers using strong encryption. SSL 3.0 was used by the IETF as the basis for developing a standard protocol to duplicate SSL functionality—known as Transport Layer Security or TLS.

SSL 3.0 and TLS 1.0 have slight differences. TLS uses HMAC algorithms for integrity checking, which is harder to break than the MAC used in SSL. But because the differences in these two protocols are negligible we will refer to the SSL/TLS protocol in this paper.

SSL/TLS does not replace FTP the way that SSH does; rather SSL/TLS can be used to create secure encrypted tunnels for FTP traffic. Two RFCs describe how to handle this process:

- FTP over SSL (RFC 2228), which extends the FTP protocol to provide strong authentication, integrity, and confidentiality on both the control and data channels.
- TLS (RFC 2246), which defines the protocol designed to allow client/server applications to communicate over the Internet without eavesdropping, tampering, or message forgery.

Something to consider when pushing FTP through an SSL tunnel is the dual-channel nature of the FTP protocol. FTP requires two tunnels—one for the control channel and one for the data channel—which means you need to open two ports in your firewall. AttachmateWRQ solutions include a "firewall-friendly" way of sending FTP traffic over SSL that requires only one port to be open.

AttachmateWRQ security solutions

AttachmateWRQ security solutions include both the SSH and SSL/TLS protocols:

- **Reflection for Secure IT**
  SSH servers for Windows, UNIX, and Linux as well as SSH thick clients for Windows, UNIX, and Linux.
- **Reflection for the Web**
  SSL/TLS thin clients for any platform and SSL/TLS proxy server for any platform.
- **EXTRA! X-treme**
  SSH and SSL/TLS thick clients for Windows.
- **Reflection emulation software**
  SSH and SSL/TLS thick clients for Windows.

Table 1 provides a quick way for you to see which AttachmateWRQ products provide the security measures you need.

Table 1: AttachmateWRQ Security Solutions

	AttachmateWRQ SSL/TLS solutions	**AttachmateWRQ SSH solutions**
Client connectivity		
On demand via a browser	Yes Thin clients work with any browser.	No Thin client support for sftp not yet available.
Command-line interface	Yes	Yes Available with both scp and sftp.
Graphical interface	Yes	Yes Available with sftp.
Interoperability	Yes Support for RFC 2228 and RFC 2246.	Yes Support for RFC 4251.
Data confidentiality		
Encryption	Yes Choice of cipher strength, including AES and 3DES.	Yes Choice of cipher strength, including AES, 3DES, and Blowfish.
SSL/TLS	Yes Support for SSL 3.0 and TLS 1.0.	Not applicable
SSH	Not applicable Support for SSH2.	Yes
Data integrity		
Message remains unchanged	Integrity checking through hashed message authentication codes.	Integrity checking through hashed message authentication codes.
Automatic checkpoint restart	No	Yes

continued on next page

Table 1: AttachmateWRQ Security Solutions (continued)

	AttachmateWRQ SSL/TLS solutions	AttachmateWRQ SSH solutions
Authentication		
User name/password	Yes	Yes
Public keys	No	Yes Can be used instead of user name and password; key agent for SSO to multiple servers; authentication of both server and client.
Digital certificates	Yes Support for x.509 certificates; cert-signing and cert-generating tool comes with the management server.	Yes Support for x.509 certificates.
Kerberos	Yes	Yes
Automation		
Scriptable	Yes	Yes
Scheduled	Yes Through external program.	Yes Through external program.
Access control		
Ability to limit access to specific directories	No	Yes Through chrooting and virtual directories.
Ability to set access rights per user	Yes Through management server and secure token authentication feature.	Yes Users can be restricted to sftp only; servers enforce security permissions for read/write access to each file and folder.
Firewall friendly	Yes With the SSL proxy server, only one port needs to be open in the firewall.	Yes Only one port is open in the firewall.
Auditing		
Logging	Yes Ability to set various logging levels; metering server provides centralized logs and reporting.	Yes Ability to set various logging levels.
Centralized management		
Centralized configuration of connections	Yes Client configurations through the management server.	Yes Client configurations through the management server.
Centralized user-access control	Yes Through configuration in the management server.	Yes Through configuration in the management server.
Cross-platform support	Yes Thick clients for Windows; thin clients run in a browser on any platform; SSL proxy server runs on any platform with a JVM.	Yes Thick clients for Windows, UNIX, and Linux; servers run on Windows, UNIX, and Linux.
End-to-end security	Yes SSL proxy server can run on the target server or can use SSL to tunnel from proxy to target server.	Yes SSH servers generally run on target servers.

SSH and SSL/TLS: Your Safest Alternatives

Although FTP is a tried and true method of delivering files over the Internet, its use can seriously jeopardize the security of your organization’s private information. Replacing or augmenting FTP with the industry-standard SSH and SSL/TLS protocols is a practical, reliable, and affordable alternative. AttachmateWRQ offers a variety of client and server security options that are based on these protocols. With our products, which are designed to work with both existing and planned security policies, you can effectively ensure the safety of data in motion.

About AttachmateWRQ

AttachmateWRQ, the leader in universal host access and integration, helps you maximize the value of your existing IT investments as you advance your long-term business strategies. More than 40,000 customers, representing over 16 million desktops worldwide, use AttachmateWRQ products and services to extend, manage, and secure their enterprise assets. Learn more at **www.attachmatewrq.com**.

attachmateWRQ™

Corporate Headquarters

1500 Dexter Avenue North
Seattle, Washington 98109

TEL 206 217 7500
800 872 2829
FAX 206 217 7515

EMEA Headquarters
The Netherlands

TEL +31 71 368 1100
FAX +31 71 368 1181

Asia Pacific Headquarters
Australia

TEL +61 3 9825 2300
FAX +61 3 9825 2399

Latin America Headquarters
Mexico

TEL +52 55 5658 7755
FAX +52 55 5658 6393

WEB
Attachmatewrq.com

E-MAIL
info@attachmatewrq.com

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Attachmate Corporation, dba AttachmateWRQ. AttachmateWRQ assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this document. AttachmateWRQ, the AttachmateWRQ logo, and Reflection are trademarks or registered trademarks of Attachmate Corporation. All other trademarks, trade names or company names referenced herein are used for identification only and are the property of their respective owners. Except as may be expressly stated in this document, any use of non-Attachmate Corporation trademarks in this document is not intended to represent that the owners of such trademarks sponsor, are affiliated with, or approve products from AttachmateWRQ. Printed in the USA.